A Secure Robust Image Steganographic Model *

Yeuan-Kuen Lee and Ling-Hwei Chen
Department of Computer and Information Science
National Chiao Tung University, Hsinchu 30050, Taiwan, R.O.C.
email: yklee@debut.cis.nctu.edu.tw, lhchen@cc.nctu.edu.tw

Abstract

The prisoners' problem is a typical model of steganography, in which two persons attempt to communicate covertly without alerting the warden. That is, only the receiver knows the existence of message sent by the sender. One available way to achieve this task is to embed the message in an innocuous-looking cover-media. On the other hand, for avoiding the covert communication taking place, the warden may be allowed to slightly modify messages as they are sent between prisoners. In this paper, we will propose an image steganographic model in which the hidden message can survive the image modification. Experimental results show these created stego-images are innocuous looking and the hidden message is robust enough to resist against some image processing operations.

Keywords: steganography, cover-image, stego-image, active warden.

1 Introduction

Steganography is an ancient art of conveying messages in a secret way such that only the receiver knows the existence of messages [1]. The techniques of steganography are classified into linguistic steganography and technical steganography [2]. The former consists of linguistic or language forms of hidden writing. The later, such as invisible ink, try to hide messages physically. One disadvantage of linguistic steganography is that users must equip themselves to have a good knowledge of linguistry. In recent years, everything is trending toward digitalization. And with the development of the Internet technologies, digital media can be transmitted conveniently over the network. Therefore, messages can be secretly carried by digital media by using the steganography techniques, and then be transmitted through the Internet rapidly.

*This work was supported in part by the National Science Council, R.O.C., under Contract NSC 89-2213-E-009-016, NSC 89-2213-E-216-031.

The steganographic terminologies used in this paper agreed with those in [3]. The goal of steganography is covert communication. So, a fundamental requirement of a steganographic system is that the hidden message carried by stego-media should not be sensible to human beings. The most general steganographic model presented by G.J. Simmons is the prisoners' problem [4]. In this problem, two persons in the jail plan to make an escape together. All communications between them are monitored by the warden. So they must hide the messages concerning escape plan in another innocuous-looking media. An assumption in this model is that both the sender and receiver must have shared some secret information before imprisonment. So the prisoners' problem is classified into secret key steganography. Pure steganography means that there is none prior information shared by two communication parties. If the public key of the receiver is known to the sender, the steganographic protocol is called public key steganography [5].

The warden may be passive, that is, he only observes the passing messages. If the warden detects the occurrence of covert communication, the prisoners will be frustrated in their attempt to escape and will be thrown into solitary confinement. Steganalysis is the art of discovering the existence of hidden information. In [6], Johnson and Jajodia identify some characteristics of stego-images that are created by specific image steganographic systems.

To remove all possible covert messages, an active warden may be allowed to slightly modify the data being sent between prisoners. An example of mild modification performed by the active warden is to replace the words with some close synonyms in the mail documents. If the carrier of secret messages is an image, any low-pass filters can be utilized for obviating covert communication. It is worthy to note that the primary goal of an active warden is to avoid covert communication taking place. On the other hand, in the real world a passive warden or monitor makes an attempt to find unknown criminals from their communication to a known criminal. Opposite to the goal of steganalysis, the requirements of a steganographic system include not only imperceptibility but also undetectability by any steganalysis tool. When examined by an active warden, the hidden message should be robust against any possible modification. There are some steganographic protocols in the presence of a passive warden or an active one are described in [7-8].

The most common and simplest image embedding method is the least significant bit (LSB) insertion. The LSB insertion embeds message in the least significant bit of some selected pixels. In this scheme, the embedding capacity can be increased by using two or more least significant bits. At the same time, not only the risk of making the embedded message statistically detectable increases but also the image fidelity degrades. Hence a variable-sized LSB embedding scheme is presented in [9-10], in which the number of LSBs used for message embedding/extracting depends on the local characteristics of the pixel.

The implementation of LSB scheme must adapt to diverse media file formats. There are many image steganographic techniques that strongly correlated with the format of the cover-image. S-Tools [11], Stego [12], and Fridrich's method [l3] are designed for palette-based images, and Jpeg-Jstego [14] is used for JPEG compressed images. Consequently, format convertion of these stego-images will destroy the whole hidden message. And slightly modify the palette in the pallette-based image or recompress the JPEG images using different quality, i.e., change the quantization table, will destroy the whole hidden message too.

The advantages of LSB-based method are easy to implement and high message payload. Unfortunately, the hidden message is vulnerable to even a slight modification from a active warden. Marvel et al. [15] present an image steganographic method, entitled spread spectrum image steganography (SSIS), that hides and recovers the message within digital imagery. The SSIS incorporated the use of error-control codes to correct the large number of bit errors. The performance of SSIS has been demonstrated by adding low levels white Gaussian noise and by applying low levels of JPEG compression. With Q-factor of 80, the resulting embedded signal BER is 0.3001. Consequently, the (2040, 32) binary expansion of Reed-Solomon code must be used for error-free message recovery.

In the next Section, we will describe our proposed blind secret-key image steganographic model. Some experimental results and performance analysis will be presented in Section 3 to show that the proposed model is practicable. Finally in Section 4, conclusions and future works will be presented.

2 The Proposed Secret-key Image Steganographic Model

In this Section, we will propose a secret-key image steganographic model. While examined by a passive warden, the message hidden in the stego-image is not only imperceptible but also undetectable. And while investigated by an active one, the hidden message is also robust enough to resist against lawful slight modifications. With the secret-key only, the original cover-image is not needed to recover the embedded message from the received stego-image.

In the proposed steganographic model, the active warden is assumed to be allowed to modify the stego-images using some image processing operations, such as lossy compressing, compression quality-factor alteration, format convertion, palette alteration, low-pass filtering. In order to survive these operations, the embedding method must embed the message in the content of the image and should be independent of the image file format.

Our embedding scheme is based on the observation that the mean value of a block is a robust attribute in the stego-images. If the message is embedded in this attribute, it can meet the robustness requirement of the steganographic model. First, we generate a

table whose entry is a binary symbol that indicates the embedding message. To increase the robustness of the embedded message, the neighboring entries are grouped together and matched to the same symbol. In the embedding module, the mean of each block are changed to the center of a group, i.e., interval. So, if the mean of a block is still locate in the same interval under some image processed operations, the embedded message can be extracted correctly.

Fig. 1 shows the block diagram of our proposed secret-key image steganographic model. The input messages can be in any digital form, and are often treated as a bit stream. Since some content-dependent patterns in the original message may reveal the existence of the hidden message, and embedding more bits of message will introduce more degradation, a compression module is used to deal with these problems. In general, the secret key is seeded into a pseudo random number generator (PRNG) to locate the embedding positions randomly. Accordingly, without the session key, neither to extract the hidden message nor to prove it's very existence is possible. In fact, embedding the message randomly is functionally similar to first permute the message, and then embeds sequentially the message in the cover-media. So, a suitable encryption scheme is necessary to be applied on the compressed message to raise the steganographic security level. Note that the encryption module can conceal the content-dependent property of the message, and the compression module should be performed prior to the encryption module for the benefit of entropy coding.

The encrypted message is then encoded in the ECC coder in order to correct errors which are caused by the modification of an active warden in the embedded message. Any error-control code [16] that is capable of correcting the bit error rate (BER) can be used in our model. Subsequently, the encoded message is permuted before embedding. This step , which is to avoid a group or burst of errors, is essential to the ECC coder. There are two stego-keys used in the proposed model, one for permuting the message, another for gerneating stego-tables. Stego-tables are both used to decide how much the mean value of a block should be changed in the embedding module, and used to judge what message is embedded in the extracting module. In consideration of user friendly, it is recommended that all the keys (include the encryption key) should be incorporated in one longer stego-key, and that does not degrade the security level of the proposed steganographic scheme.

When the receiver receives a stego-image, he must use the same stego-key to generate these stego-tables to extract the embedded message. Then, the extracted message should be inversely permutated, ECC decoded, decrypted, and decompressed in sequence. Finally, the receiver can see the hidden information.

In the following subsection, we will describe the stego-table generation module, the preprocessing module, the embedding module and the extracting module in more detail.

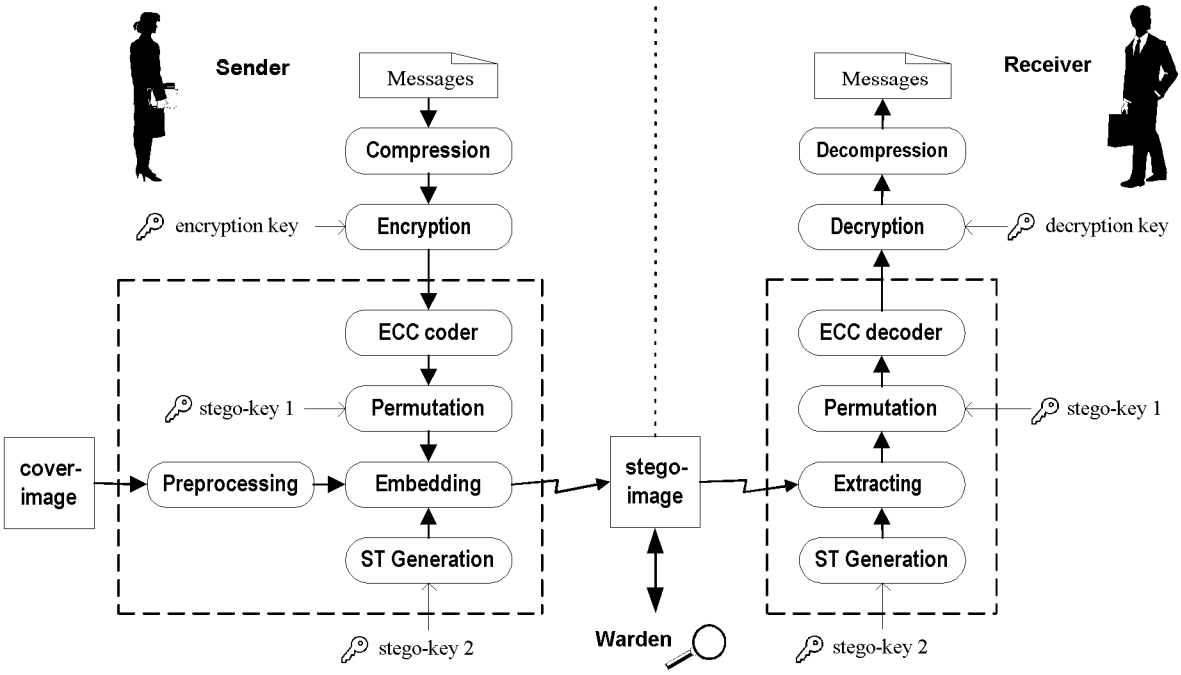

Figure 1: The block diagram of the proposed secret-key image steganographic model.

2.1 Stego-table Generation

Stego-tables, used both in the embedding module and in the extracting module, contain information indicating what message is embedded in the mean value of a block. Except for boundary areas, the numeral spectrum of mean value is divided into many equal-sized intervals. The mean value of each block will be adjusted to the center of a certain interval. If the mean value of a block is not changed to another interval, the embedded message can be extracted correctly in the extracting module. Hence the length of the interval determines the tolerance to the modification by an active warden.

Stego-tables are generated by a PRNG which is seeded with the shared stego-key. In order to avoid generating consecutive 1's or 0's, each output symbol 1 is expanded into 10, and 0 is expanded into 01. For example, if the output sequence of the PRNG is 11010001, it will be expanded into 1010011001010110. Note that it is guaranteed that each symbol can find it's complement in it's neighbors. Finally, these symbols will be assigned to the intervals sequentially.

In a natural image, the mean values of all blocks may widely spread all over the numeral spectrum. If only one stego-table is used in the proposed scheme, all of the block means in the created stego-image will occur near the centers of intervals. This is a specific characteristic that may reveal the existance of the hidden message and defeat the purpose of steganography. So, at least k stego-tables are necessary to fade this characteristic when the length of an interval is k. Let ST_i denote the i-th stego-table, $i = 0, 1, ..., k-1$. The center of each interval in ST_i will then locate at $nk + i$, where n is a positive integer.

Therefore, all the possible mean values can be selected in the embedding process.

Up to now, a new problem should be sloved is that which stego-table will be chosen for embedding the hidden message for a certain block. When the PRNG has finished generating all stego-tables, we assign the next task to generate a sequence of table indices for every block in the image. Every 8 consecutive binary symbols derived from the PRNG will be combined together to produce a numeral, and then using a modulus operation (mod k) to generate the table index.

2.2 Preprocessing

In the embedding module, all of gray scales of the pixels within a block will be modified such that the mean values of the block will located at the center of the interval. If there are too many pixels whose gray scales are near the gray scale boundary, i.e.,0 or 255, it will be hard to move the mean of a block toward the boundary while maintaining the image fidelity. Thus, a compaction process is used to perform initial processing such that all gray scales will be away from the boundary.

Let $f(x, y)$ represent the original cover-image. The preprocessing procedure is to generate a compacted image $f'(x, y)$ whose gray scale at every point (x, y) is in the interval $[k, 255 - k]$. Thus, the compaction process can be done with the following equation:

$$f'(x, y) = k + f(x, y) \times \frac{(128 - k)}{128},$$

where k is the length of the interval. Note that if k is not too large, $f'(x, y)$ will be indistinguishable from $f(x, y)$ by the naked eye.

2.3 Embedding Process

Using the stego-tables and a sequence of table indices, the permuted message is embedded in the compacted cover-image. The compacted cover-image $f'(x, y)$ is divided into blocks of $n * n$ pixels. For each block, one message-bit is embedded in a block from left to right and top to bottom.

Let B_c be the current processing block, and B_u, B_l be the upper and left block of B_c, respectively. Let m_c, m_u and m_l be the destination means of B_c, B_u and B_l, respectively. Given a message-bit b and a table index i, the mean, denoted by μ, and standard derivation, denoted by σ, of B_c are first computed. Then the stego-table ST_i is searched for the closest interval center, denoted by c_1, with the interval value same as b. If both m_u and m_l are larger than μ, the second closest interval center, denoted by c_2, is searched under the following condition:

$$(c_2 > \mu)\, and\, ((c_2 - \mu) \leq 3k).$$

If both m_u and m_l are smaller than μ, c_2 is searched under the following condition:

$$(c_2 < \mu)\, and\, ((\mu - c_2) \leq 3k),$$

where k is the length of a interval. If c_2 can not be found, c_1 is assigned as the destination mean of B_c. If c_2 is found, m_c is assigned by the following expression:

$$m_c = \begin{cases} c_2, & \text{if } |c_2 - m_u| + |c_2 - m_l| < |c_1 - m_u| + |c_1 - m_l|, \\ c_1, & \text{otherwise.} \end{cases}$$

When m_c has been determined, the next step is to adjust the gray scale of every pixel in B_c such that μ will approach m_c. There are two simple methods to achieve this task. One is to uniformly change the gray scale of each pixel in B_c, the other is that the gray scale is proportionally changed. The advantage of the proportional method is that the blocking effect is less noticeable. However, more degradation will be introduced in the edge blocks that are located on the boundary of a large smooth region. So a hybrid method is used in our embedding scheme. The uniform method is used only in these edge blocks, where their standard derivations are larger than a threshold value T. The other blocks will then be embedded using the proportional method. To further reduce the blocking effect, a white Gaussian noise with zero-mean and variance s, is added to the image. Note that adding the Gaussian noise will not influence the mean of a block in theory. In other words, the embedded message will not be destroyed by the Gaussian noise adding process. Let $\eta(x, y)$ be the additive Gaussian noise function, and $g(x, y)$ be the created stego-image. Then, $g(x, y)$ can be obtained according to the following expression:

$$g(x,y) = \begin{cases} f'(x,y) + (m_c - \mu) + \eta(x,y), & \text{if } \sigma > T, \\ f'(x,y) + \frac{n^2(m_c - \mu)f'(x,y)}{\sum_{(x,y)\in B_c} f'(x,y)} + \eta(x,y), & \text{otherwise.} \end{cases}$$

where $f'(x, y)$ is the compacted cover-image, m_c is the destination mean of B_c, μ is the mean value of B_c, σ is the standard derivation of B_c, T is a threshold value.

When the sender has finished the embedding process, the created stego-image $g(x, y)$ are then sent to the reciver. The extracting process done by the receiver will be described in the next subsection.

2.4 Extracting Process

When the receiver receives a stego-image, the same stego-keys are used to generate the same stego-tables and table indices. The received stego-image $g'(x, y)$, which may be slightly modified, is then divided into blocks of $n * n$ pixels. For every block, the mean value is computed and then used to find the embedded message from the corresponding stego-table. Given a table index i, the embedded message b can be obtained by the following equation:

$$b = ST_i(\mu)$$

(a) (b)

Figure 2: An embedding result of the proposed hybrid method. (a) A cover-image, entitled Lenna ($512 * 512$). (b) The created stego-image ($k = 3, s = 2.5, T = 25$).

where ST_i is the i-th stego table, μ is the mean value of the current processing block. All the extracted message are then inversely permuted, ECC decoded, decrypted, and decompressed in sequence. Finally, the receiver can see the hidden information.

3 Experimental Results and Performance Analysis

We have tested the proposed embedding scheme on a lot of gray-scale images. Fig. 2 shows an embedding result of the proposed hybrid method. Fig. 2(a) is a gray-scale cover-image of size $512 * 512$ pixels, entitled Lenna. The size of a block is $8 * 8$ pixels, so there are total 4096 message-bits embedded in Fig. 2(b). The length of interval was chosen to be 3, the variance of Gaussian noise was given by $s = 2.5$, and the standard derivation threshold $T = 25$.

In our experiments, the robustness was tested using gray scale manipulations, such as JPEG encoding/decoding, low-pass filtering, and noise adding. Most of the image processing operations were carried out in Photoshop 5.5. Table 1 displays the robustness of the proposed embedding scheme to JPEG compressions. As can be seen, the proposed embedding scheme perform very well and achieve 0% BER for quality factor as low as 3.

The embedded message BER values to noise adding and low-pass filtering are given in Table 2. Most embedded message BER values are very low, so many proper error control codes can be chosen to exactly correct the whole embedded message. The maximum BER value, (BER=18.75%) occurred when the amount of Gaussian noise was chosen to be 10,

nevertheless it still within the acceptable BER range (BER=22%) for the 1/6 convolution code.

4 Conclusions and Future Works

In this paper, we have introduced a secure robust image steganographic model which can escape the notice of a passive warden and is resistant to the slight modification done by an active warden. The proposed embedding scheme is based on the observation that the mean value of a block is a robust attribute, so the hidden message is edbedded in these mean values. The numeral spectrum of the mean value is divided into many equal-sized interval, and a PRNG seeded with a stego-key is used to assign each interval a binary symbol. Using the proposed hybrid method, the mean value of each block is adjusted to a certain interval center while maintaining the image fidelity and reducing the blocking effect. To escape the notice of a passive warden, at least k (interval length) stego-tables and a PRNG are used to fade a specific pattern, which may reveal the existence of the hidden message. Experimenal results show that the embedded message may be survivable under slight modifications, such as JPEG compression, noise adding, low-pass filtering. Since the embedded message BER is small enough to be corrected by a suitable error control code, the receiver can see the whole hidden information sent by the sender. Note that the more powerful capability of error-control code is used in the steganographic model, the lower payload of message is. This means that how to choose a proper error control code is important in the steganographic field. This is the future work in our study.

Table 1: Robustness to JPEG compression.

Quality	bits per. pixel	No. of error bit	Embedded message BER
4	0.83	0	0.00%
3	0.71	0	0.00%
2	0.54	6	0.15%
1	0.44	394	9.62%

Table 2: Robustness to noise adding and low-pass filtering.

Slight modification	No. of error bit	Embedded message BER
Uniform noise adding (amount=5)	1	0.02%
Uniform noise adding (amount=10)	178	4.35%
Gaussian noise adding (amount=5)	50	1.22%
Gaussian noise adding (amount=10)	786	18.75%
Blur	0	0.00%
Blur More	71	1.73%
Median filtering (3*3 mask)	120	2.93%
Mean filtering (3*3 mask)	123	3.00%

References

[1] David Kahn, "The Codebreakers: The Comprehensive History of Secret Communication from Ancient Times to the Internet," Scribner (1996).

[2] F. L. Bauer, "Decrypted Secrets - Methods and Maxims of Cryptology," Berlin, Heidelberg, Germany, Springer-Verlag (1997).

[3] Birgit Pfitzmannn, "Information Hiding Terminology", in Proceedings of First Workshop of Information Hiding, Cambridge, U.K. May 30 - June 1, 1996. Lecture Notes in Computer Science, Vol.1174, pp 347-350. Springer-Verlag (1996).

[4] Gustavus J. Simmons, "The Prisoners' Problem and the Subliminal Channel", in Proceedings of CRYPTO '83, pp 51-67. Plenum Press (1984).

[5] Stefan Katzenbeisser and Fabien A. P. Petitcolas, "Information Hiding Techniques for Steganography and Digital Watermarking," Artech House (2000).

[6] Neil F. Johnson and Sushil Jajodia, "Steganalysis of Images Created using Current Steganography Software," in Proceedings of 2nd International Workshop on Information Hiding, April 1998, , Portland, Oregon, USA. pp. 273 - 289.

[7] Ross J. Anderson and Fabien A.P. Petitcolas, "On the limits of steganography," IEEE Journal on Selected Areas in Communications (J-SAC), Special Issue on Copyright & Privacy Protection, vol. 16 no. 4, pp 474-481, May 1998

[8] Scott Craver, "On Public-key Steganography in the Presence of an Active Warden," in Proceedings of 2nd International Workshop on Information Hiding, April 1998, Portland, Oregon, USA. pp. 355 - 368.

[9] Yeuan-Kuen Lee and Ling-Hwei Chen, "An Adaptive Image Steganographic Model Based on Minimum-Error LSB Replacement," in Proceedings of the Ninth National Conference on Information Security, pp. 8-15. Taichung, Taiwan, May 14-15, 1999.

[10] Yeuan-Kuen Lee and Ling-Hwei Chen, "A High Capacity Image Steganographic Model," accepted by IEE Proceedings Vision, Image and Signal Processing. (2000)

[11] A. Brown, "S-Tools", Shareware
ftp://idea.sec.dsi.unimi.it/pub/security/crypt/code/s-tools4.zip (Version 4)

[12] Romana Machado, "Stego", Shareware, http://www.stego.com

[13] Jiri Fridrich, "A New Steganographic Method for Palette-Based Images", in Proceedings of the IS&T PICS Conference, Savannah, Georgia, April 25-28, 1999, pp.285-289.

[14] D. Upham, Jpeg-Jstego, Modification of the Independent JPEG Group. JPEG software (release 4) for 1-bit steganography in JFIF output file.
ftp://ftp.funet.fi/pub/crypt/steganography

[15] Lisa M. Marvel, Charles G. Boncelet, Jr. and Charles T. Retter, " Spread Spectrum Image Steganography," IEEE Transaction on Image Processing, August 1999, Vol. 8, NO. 8, pp. 1075-1083.

[16] Shu Lin and Daniel J. Costello, Jr. "Error Control Code: Fundamentals and Applications," Prentice Hall (1983).